P&I Functional Model (DRAFT)

Overview

This document provides a general description of the components and functions of the provisioning and integration component of an institutional-scale Identity and Access Management (IAM) suite. It also suggests touch points with other subsystems in such a suite. Requirements for provisioning and integration functionality and operation can be written based on the terms and concepts presented in this model.

The aim of the CIFER provisioning and integration facilities is to provide the tools and services necessary to integrate the identity information and services offered by the rest of the CIFER IAM suite with a variety of real-world identity consumers (on-prem applications, cloud-based services, and, in some cases, even other IAM tools and services). In short, the facilities provide mechanisms whereby the valuable identity information constructed and managed within other CIFER components can be made useful within non-CIFER contexts.

In some cases, identity integration may be accomplished directly with other CIFER components; for these cases, the provisioning and integration facility includes recipes and exemplars to aid consumers in taking advantage of CIFER facilities. In other cases, consumers may require translation services and business logic capabilities to address structural differences between their native identity strategies and those used within the CIFER suite; for these cases, the provisioning and integration facility provides a collection of services, from schema translation facilities to business logic modules, suited to a variety of integration models. In still other cases, consumers may have their own, embedded identity ecosystems which need to be populated and maintained in a fashion that keeps their representations of identities consistent with those managed by CIFER components; for these cases, the provisioning and integration facility provides tools to support a variety of provisioning approaches -- from traditional batch update models through more contemporary, event-driven provisioning via standard APIs (eg., SCIM) and messaging facilities (eg., JMS).

Terms and Definitions

Identity management systems like those included in the CIFER suite typically construe **identities** as electronic representations of those aspects (often referred to as "**attributes**") of an entity which are of importance to an organization. The most commonly noted entities are people associated in some fashion with organizations, but more abstract entities may also need to be represented by electronic identities -- groups, organizations, services, all may need to be represented electronically, and all may have (albeit, in many cases, domain-specific) associated attributes. The identity management components of CIFER -- the CIFER registry, for example -- are largely concerned with *establishing* and *maintaining* identities over time within an organization.

Identity **consumers** are entities (often applications and software systems, but sometimes individuals or organizations) that use electronic identities. Use cases range from the casual (as with a cloud-based application that offers authenticated users the ability to customize and store UI preferences) to the mission-critical (as with an HR system that must route payroll change orders through multiple levels of approval based on the relative roles of the actors involved in a complex hiring scenario).

Systems of record are entities (usually software systems, but in some scenarios, human actors) that provide authoritative information about entities to IAM facilities. Organizations typically operate multiple systems of record for different purposes (student information systems, HR systems, financial systems, etc.). One of the unique factors making higher ed institutions ill-suited to use many commercial IAM suites is the extent to which our users typically have aspects of their identities sourced from multiple systems of record. It is possible for a system to be both a system of record and an identity consumer -- a student information system may act as a system of record for student identities, but as an identity consumer for faculty identities, or a faculty activity tracking system may act as a system of record for faculty appointments, but as a consumer of faculty HR information.

Some identity consumers do not maintain any persistent store of identities, themselves, relying solely on external IAM facilities in realtime to fulfill their identity needs. Many other consumers maintain some persistent store of identity information -- account databases, user profiles, application role repositories. We refer to the processes by which identities in these latter consumers' contexts are made and subsequently kept consistent with identities managed within central IAM facilities as **provisioning**.

Some consumers are designed so as to require the existence of a context-specfic identity prior to an entity's first interaction with them -- an LMS system, for example, may require the presence of a student's identity in its user repository in order for an instructor to assign the student access to course materials even before the student's first login session. We refer to the **provisioning style** that maintains consistent identities for entities which may not yet be engaged with consumers as **just in case provisioning**, since its aim is to provision identities *just in case* they're needed by a consumer. Other consumers are designed so as to establish identities at the moment entities first present to them in need of their services -- many social networking and cloud-based services operate in this mode. We refer to the provisioning style that establishes consistent identities in the context of consumers on demand, when entities first interact with them as **just in time provisioning**, since its aim is to effect provisioning *just in time* for the consumer to fulfill its identity requirements.

Identity consumers may support different **provisioning models**, largely dependent on their expectations regarding the fluidity of identity information and their level of maturity with respect to identity management. Provisioning models span continua from very high to very low latency and from very low to very high complexity. Typical provisioning models mirror, in many respects, classical enterprise integration patterns.

Components of a Provisioning Facility

Authortative Source Integration

One of the primary functions of a provisioning facility must be integration with one or more authoritative sources of identity information. In the case of CIFER, the primary identity authority is the Identity Registry. In order to establish and maintain the consistency of identity states between identity consumers and identity registries, provisioning facilities must be integrated with those registries. This integration could be accomplished through a variety of mechanisms, depending on the services exposed by each registry. In all cases, however, provisioning facilities must be able to both retrieve identities from the registry and identify changes to identities (creation, deletion or archiving) and their attributes (creation, modification, removal). Integration can be achieved either through registry-initiated means (as would be the case if the registry exposes a real-time changelog or provides a messaging-based change notification service) or through provisioning facility-initiated means (as would be the case if the registry exposes a RESTful query interface representing identities as resources, for example, or a SQL query interface for retrieving identity information). The CIFER provisioning facility aims to be flexible enough to support a variety of methods for integration with identity sources, but at a minimum requires the ability to initiate retrieval of identity information from the registry.

Consumer Authorization and Consumer Customization

Not all provisioning-reliant consumers represent the full gamut of identity information housed in identity registries within their own identity stores. Individual consumers may need only a subset of the identities housed in an identity registry reflected in their own identity stores, and may only support a subset of the attributes comprised by those identities within the identity registry. Further, there may be institutional policies (or even legislation) mandating that certain identity information be withheld from certain consumers. For this reason, an effective provisioning facility needs the ability to store and act upon consumer-specific configuration information characterizing both the needs of and the limitations to be enforced against specific consumers with regard to the provisioning of identities to them. This configuration information needs to be applied regardless of the provisioning model employed with any individual consumer and regardless of the provisioning style implemented for a particular consumer. In some cases, consumer authorization and subscription rules may be explicit; in other cases, they may be implicit or inherited from default configuration parameters.

Attribute Transformation and Consumer Translation Business Logic

If all identity consumers shared a single perspective on identity, using a common schema, common data dictionaries, etc. for identity information, there might be little need for business logic within a provisioning suite. Identity consumers within our institutions vary widely in their approaches to identity, however, and often expose no interfaces of their own for "impedance matching" with identity infrastructures using different identity schemas. For this reason an effective provisioning facility needs the ability to express, manage, and act on business logic of a number of types:

- **Attribute Name Mappings.** Many consumers may share common identity attributes with a central IAM facility, but may not share common naming conventions for them. An effective provisioning facility will expose a mechanism for implementing attribute name mapping (eg., "givenName" -> "firstName") on a per-consumer basis to address this situation.
- **Attribute Format Transformations.** Consumers may have conflicting requirements for the format of certain identity information. One consumer may, for example, require dates to be presented and stored in some epochal form, while another may require the same dates to be expressed in MM-DD-YYY format. To address these differences, an effective provisioning facility will need to expose mechanisms for implementing reformatting of values as well as attribute names.
- **Selection/Promotion logic.** Some consumers may vary in their definitions of certain key identity attributes -- one consumer may consider organizational unit affiliation to be a single-valued attribute, while another may support multiple departmental affiliations. A full-featured provisioning facility will provide means for resolving such value-selection and promotion conflicts on a per-consumer basis.
- **Typographical Construction.** A common requirement of certain identity consumers is the construction of synthetic identity attributes from component attributes stored in a central IAM facility -- an IAM facility may, for example, store multiple parts of an individual's name, while a particular consumer may require a single "full name" presentation. In order to support such consumers, a full-featured provisioning facility should provide a means to implement such typographical constructions as part of the business logic associated with individual consumers.

Consumer Presentation and Adapters

Identity consumers fall along a wide spectrum in terms of their tolerance for latency (i.e., how long the state of an identity in their context may be consistent with a *prior state* of the corresponding identity in their associated IAM facilities) and their ability to maintain identity information dynamically (ie., how frequently/quickly they can process changes to identity information). To support the wide range of identity consumers deployed on our campuses, not to mention the plethora of consumers with varying requirements available via cloud providers, an effective provisioning facility needs the ability to support a range of provisioning models, from the most basic to the most sophisticated. We identify the following provisioning models as being important for any effective provisioning facility to support:

- **Full batch import.** In this model, the provisioning facility periodically constructs a formatted extract containing the attribute information and identities relevant for (and authorized to) a given consumer, and makes the extract available for import by the consumer. The consumer, in turn, replaces its entire identity state with the "snapshot" provided by the provisioning facility. This is typically the highest-latency of the core provisioning models, and among the simplest to implement and maintain.
- *Incremental batch import. *Some consumers can additionally support incremental batch processing of identity information. In this model, the provisioning facility periodically constructs a formatted extract containing a compendium of the changes made to the states of identities in the central IAM facility since the last time an extract was produced, and delivers this compendium of changes to the consumer. The consumer, in turn, applies the changes indicated in the extract to bring its data into a state consistent with the state of the central IAM facility. This model typically can provide somewhat lower latency than the full batch import model, but it does so at the expense of being significantly more complicated. This model is often implemented in conjunction with the full batch import model to address some of its intrinsic weaknesses.
-

<models and APIs - from a bulk identity extractor through SCIM client interface, SCIM server interface, event notification + MQ>

(KHazelton 8-Oct-2013): For a list of the modes of consumer presentation, see the Provisioning and Integration section of the capability "wish list". I'd welcome a review of the list to see if we have agreement on what's on it and that nothing important is missing.

Provisioning Scenarios

<For each exemplar, walk through the process by which the scenario unfolds with respect to the provisioning components, mixing up the descriptions to touch on different provisioning styles and models>

<Exemplar 1: Creation of identity in a cloud application - new employee onboarding>

<Exemplar 2: Creation of identity in an on prem service - employee enrolls in a course and is provisioned to an LMS as a result>

<Exemplar 3: Faculty member marries and has name change provisioned into LMS *but not into grant management system*>

<Exemplar 4: Student asserts FERPA rights and has contact information *deprovisioned* from a wiki>

Touchpoints

<Provisioning is only one part of the whole CIFER environment, and like other components of the CIFER model, these tools require services and support from other components. Here, we'll document these touchpoints with other components/workstreams.>